CLAIMS

1. A microprocessor chip, comprising:

instruction pipeline circuitry;

address translation circuitry; and

table lookup circuitry designed to index into a table, the table having an entry associated with each corresponding address range translated by the address translation circuitry, each entry describing a likelihood of the existence of an alternate coding of instructions located in the respective corresponding address range, the table lookup circuitry further designed to retrieve a table entry corresponding to the address, the table lookup circuitry operable as part of the basic instruction cycle of executing an instruction of a non-supervisor mode program executing on a computer;

interrupt circuitry cooperatively designed with the instruction pipeline circuitry to trigger an interrupt on execution of an instruction of a process, synchronously based at least in part on a memory state of the computer and the address of the instruction, the architectural definition of the instruction not calling for an interrupt, a handler for the interrupt being responsive to the contents of the table to affect the instruction pipeline circuitry to effect control of an architecturally-visible data manipulation behavior or control transfer behavior of the instruction based on the contents of a table entry associated with the instruction.

2. A method, comprising the steps of:

as part of the basic instruction cycle of executing an instruction of a non-supervisor mode program executing on a computer, consulting a table, the table being addressed by the address of instructions executed, for attributes of the instructions;

controlling an architecturally-visible data manipulation behavior or control transfer behavior of the instruction based on the contents of a table entry associated with the instruction.

3. The method of claim 2, wherein the control of control transfer behavior includes transfer of execution control to a different instruction for execution.

4. The method of claim 3, wherein the different instruction is coded in an instruction set architecture (ISA) different than the ISA of the executed instruction.

5. The method of claim 2, wherein the control of architecturally-visible data manipulation behavior includes changing an instruction set architecture under which instructions are interpreted by the computer.

6. The method of claim 2, wherein the behavior control includes selecting between two different instruction set architectures, and the computer includes instruction pipeline circuitry designed to effect interpretation of computer instructions under the two instruction set architectures alternately.

7. The method of claim 2,
wherein each entry of the table describes a likelihood of the existence of an alternate coding of instructions located in the respective corresponding address range.

8. The method of claim 2,
wherein each entry of the table corresponds to a page managed by a virtual memory manager, circuitry for locating a table entry being integrated with virtual memory address translation circuitry of the computer.

9. The method of claim 2, further comprising the steps of:
triggering an interrupt on execution of an instruction of a process, synchronously based at least in part on a memory state of the computer and the address of the instruction, the architectural definition of the instruction not calling for an interrupt.

10. A microprocessor chip, comprising:
instruction pipeline circuitry;
table lookup circuitry designed to index into a table by a memory address of a memory reference arising during execution of an instruction, and to retrieve a table entry corresponding to the address;

the instruction pipeline circuitry being responsive to the contents of the table to affect a manipulation of data or transfer of control defined for the instruction.

11. The microprocessor chip of claim 10, further comprising:

a binary translator programmed to translate at least a selected portion of a computer program from a first binary representation to a second binary representation; and

pipeline control circuitry is further designed to initiate a determination of whether to transfer control from an execution of the first binary representation of the program to the second, and effective to initiate the determination with neither a query nor a transfer of control to the second binary representation being coded into the first binary representation.

12. The microprocessor chip of claim 10, further comprising:

interrupt circuitry cooperatively designed with the instruction pipeline circuitry to trigger an interrupt on execution of an instruction of a process, synchronously based at least in part on a memory state of the computer and the address of the instruction, the architectural definition of the instruction not calling for an interrupt.

13. The microprocessor chip of claim 12, further comprising:

interrupt handler software designed to service the interrupt and to return control to an instruction flow of the process other than the instruction flow triggering the interrupt, the returned-to instruction flow for carrying on non-error handling normal processing of the process.

14. A microprocessor chip, comprising:

instruction pipeline circuitry;

address translation circuitry; and

a lookup structure having an entry associated with each corresponding address range translated by the address translation circuitry, the entry describing a likelihood of the existence of an alternate coding of instructions located in the respective corresponding address range.

15. The microprocessor chip of claim 14, wherein the entry is an entry of a translation look-aside buffer.

20 18
16. The microprocessor chip of claim 14, wherein the alternate coding is coded in an instruction set architecture (ISA) different than the ISA of the instruction located in the address range.

17. The microprocessor chip of claim 14, wherein:

the instruction pipeline circuitry is responsive to the contents of the lookup structure entry to affect a manipulation of data or transfer of control defined for the instruction.

18. The microprocessor chip of claim 14, further comprising:

interrupt circuitry cooperatively designed with the instruction pipeline circuitry to trigger an interrupt on execution of an instruction of a process, synchronously based at least in part on a memory state of the computer and the address of the instruction, the architectural definition of the instruction not calling for an interrupt.

19. A microprocessor chip, comprising:

instruction pipeline circuitry; and

interrupt circuitry cooperatively designed with the instruction pipeline circuitry to trigger an interrupt on execution of an instruction of a process, synchronously based at least in part on a memory state of the computer and the address of the instruction, the architectural definition of the instruction not calling for an interrupt.

24 23
20. The microprocessor chip of claim 19, further comprising:

interrupt handler software designed to service the interrupt and to return control to an instruction flow of the process other than the instruction flow triggering the interrupt, the returned-to instruction flow for carrying on non-error handling normal processing of the process.

25 24
21. The microprocessor chip of claim 20, wherein the interrupt handler software is programmed to change an instruction set architecture under which instructions are interpreted by the computer.

22. The microprocessor chip of claim 20, wherein the instruction text beginning at the returned-to instruction is logically equivalent to the instruction text beginning at the interrupted instruction.

23. The microprocessor chip of claim 19, further comprising:

table lookup circuitry designed to index into a table by a memory address of a memory reference arising during execution of an instruction, and to retrieve a table entry corresponding to the address;

the instruction pipeline circuitry being responsive to the contents of the table to affect a manipulation of data or transfer of control defined for the instruction.